Arduino Platform Project

Răzvan Boboș

January 2025

1 Project Assumptions

The Trivia Game project is designed to provide an educational experience by asking players with a series of general knowledge questions via the serial connection and the D1 Shield interface.

- **Hardware Assumptions:**
  - The system will use a Arduino UNO Board along with a D1 Robot Shield provided by the university teacher.
  - The system will use a HD44780 LCD for display purposes. The LCD will display the game stages, the score and the current answers.
  - User input will be handled through a set buttons interfaced via an ADC.
  - The system will utilize UART to send trivia questions via serial communication, enhancing the interactive experience without overloading the LCD display.
- **Software Assumptions:**
  - The game will be developed using embedded C++ programming.
  - It will use the 3 presented libraries in the laboratories : HD44780, libADC, uartbuffer.
- **User Interaction Assumptions:**
  - The user interface will be intuitive enough for users without prior experience with similar devices.
- **Educational Value Assumptions:**
  - Interaction with the game will enhance users' knowledge and make learning enjoyable.

2 Project Schematic

This section outlines the hardware used in the Trivia Game project. I utilized the Arduino Uno board along with the D1 Robot Shield. The schematics of these components are important for understanding the configuration of analog inputs, the button interfaces, and the LCD screen.

2.1 Arduino Uno Board

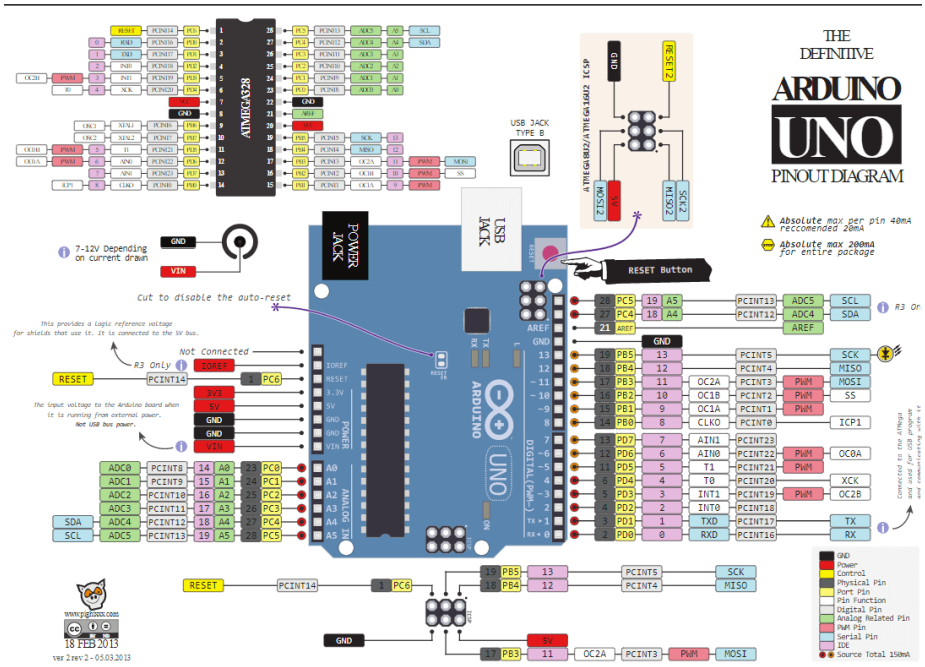

Figure 1: Arduino Uno Board Pinout Diagram

2.2 D1 Robot Shield

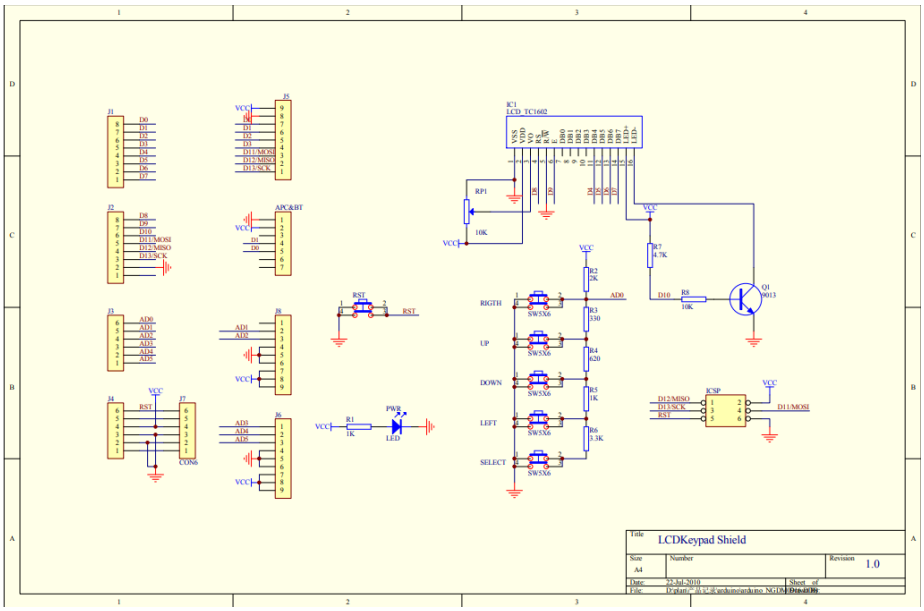

Figure 2: D1 Robot Shield Schematic

3 Implementation

3.1 Initialization

The system initializes the LCD display, UART communication, and the ADC converter.

```
#include <stdio.h>
#include <util/delay.h>
#include <stdlib.h>
#include "HD44780.hpp"
#include "libADC.hpp"
#include "uart_buffer.hpp"

int main(void) {
    LCD_Initalize();
    uart_init(9600, 0);
    ADC_Init();
    sei();  // Enable global interrupts
}
```

3.2 Data Structures for Game Content

The game involves several components of data management: questions, possible answers, and the identification of correct answers. Each component is stored in arrays for efficient access during game operation.

```
const char *questions[] = {
    "How Much Percentage Of  Earths  Surface Is Covered By Water?",
    "What was Martin Luther King Jr famous speech commonly referred to as?",
    "Who directed the film 'Schindlers List'?",
    "In the film 'The Shawshank Redemption,' the name of the main character played by Tim Robbins is?",
    "Who won the Academy Award for Best Actress for her role in the film 'La La Land'?",
    "Which is the highest-grossing animated film of all time?",
    "In which year was the first Academy Awards ceremony held?",
    "Who played the character of Jack Dawson in the film 'Titanic'?",
    "What is the capital of France?",
    "Who wrote 'Romeo and Juliet'?"};
```

```
const char *possibleAnswers[][4] = {
    {"51%", "61%", "91%", "71%"},
    {"I have a dream", "Freedom Fighters", "Dead and Evil", "Rise Up"},
    {"Steven Spielberg", "Martin Scorsese", "Michael Jackson", "Christopher Nolan"},
    {"Andy Dufresne", "Red", "Ellis Boyd 'Red' Redding", "Warden Norton"},
    {"Emma Stone", "Meryl Streep", "Jennifer Lawrence", "Natalie Portman"},
    {"Toy Story 4", "Avatar", "Frozen 2", "The Lion King"},
    {"1927", "1932", "1929", "1951"},
    {"Leonardo DiCaprio", "Tom Hanks", "Brad Pitt", "Matt Damon"},
    {"Berlin", "Madrid", "Paris", "Rome"},
    {"Charles Dickens", "William Shakespeare", "Jane Austen", "Mark Twain"}};
```

```
const char correctAnswers[] = {'d', 'a', 'a', 'a', 'a', 'd', 'b', 'a', 'c', 'b'};
```

3.3 Main Game Loop

This code continuously monitors user input from an ADC, resetting the input state when no significant change is detected, and processing the input when changes exceeds 50 units from the last reading.

```
while (1) {
    user_input = ADC_conversion();
    if (user_input > 900) { // no button pressed
        old_user_input = user_input;  // reset the input
    } else if (abs(user_input - old_user_input) > 50) { //here a change occcured
```

3.4 Game Logic

The provided code snippet is part of the main game loop for a trivia game, handling user input from ADC to determine which button was pressed and then checking whether the selected answer matches the correct answer. The ADC values are mapped to specific answers ('a', 'b', 'c', 'd'), and based on this input, the game updates the display to show whether the selected answer is correct or incorrect, updates the score, and then goes to the next question. After updating the display, the game resets the flags and increases the index.

```
if (user_input < 100)
{
    state = 'd';
}
else if (user_input < 250)
{
    state = 'b';
}
else if (user_input < 350)
{
    state = 'c';
}
else if (user_input < 500)
{
    state = 'a';
}

if (state == correctAnswers[index])
{
    count++;
    LCD_WriteCommand(HD44780_CLEAR);
    _delay_ms(10);
    LCD_GoTo(0, 0);
    sprintf(buf, "%c correct!", state);
    LCD_WriteText(buf);
}
else
{
    LCD_WriteCommand(HD44780_CLEAR);
    _delay_ms(10);
    LCD_GoTo(0, 0);
    sprintf(buf, "%c incorrect(%c)", state, correctAnswers[index]);
    LCD_WriteText(buf);
}

LCD_GoTo(1, 1);
sprintf(buf, "score: %d/%d", count, index + 1);
LCD_WriteText(buf);
_delay_ms(1000);

index++;
old_user_input = 1023;
question_sent = 0;
```

3.5 Flags and Counters Used

Here is a listing of the primary flags and counters used in the Trivia Game project to manage game flow and logic:

```
// Game control flags and counters
int game_started = 0; // Flag to check if the game has started
int question_sent = 0; // Flag ensure question is only sent once
int index = 0;         // Counter for the current question number
uint8_t count = 0;     // Counter for correct answers
```

3.6 Game End and Restart

- **Display Game Over Message:** Clear the LCD and show "END OF THE GAME!" to inform the player that the current game session has concluded.
- **Show Final Score:** Display the final score, calculated based on the number of questions correctly answered.
- **Prompt for Restart:** Prompt the player with "Press SELECT to play again," allowing them to restart the game.
- **Wait for Input:** The game enters a wait state, continuously monitoring for the SELECT button press to restart the game.
- **Reset Game Variables:** Once SELECT is pressed, game control variables are reseted.

```
else  //important code parts only
{
    LCD_WriteCommand(HD44780_CLEAR);

    sprintf(buf, "END OF THE GAME!");
    sprintf(buf, "Final sc.: %d/%d", count, index);

    _delay_ms(10);
    LCD_GoTo(0, 0);
    sprintf(buf, "Press SELECT");
    LCD_WriteText(buf);
    LCD_GoTo(1, 1);
    sprintf(buf, "to play again.");

    do
    {
        user_input = ADC_conversion();
        _delay_ms(200);
    } while (user_input > 750);
    index = 0;
    count = 0;
    old_user_input = 1023;
    game_started = 0;
}
```

4 Pictures Showing the Operation of Application

4.1 Game introduction

The initial stage of the Trivia Game introduces players to the game by displaying the game title, providing instructions on how to play and inviting them to answer the first question.

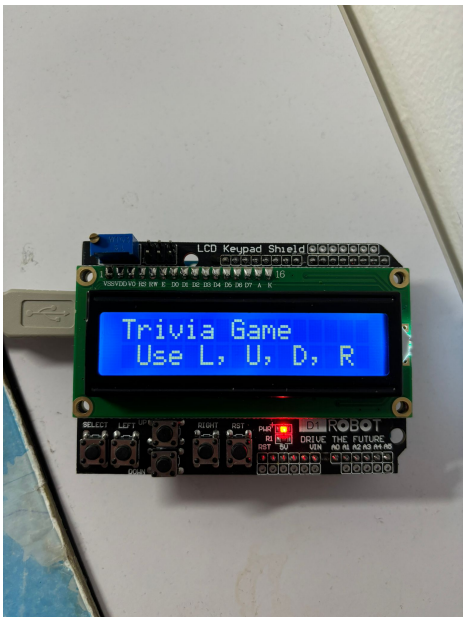

Figure 3: Game title and instructions.

Figure 4: Instructions Explanation.

Figure 5: Inviting players to answer first question.

4.2 Gameplay Screenshots

This section presents a series of images that illustrate the gameplay interface (Serial Monitor) and user interactions with the shield buttons.

```
Question no 0:
How Much Percentage Of Earth's Surface Is Covered By Water?
a) 51%
b) 61%
c) 91%
d) 71%
```

Figure 6: First Question Displayed on Serial Monitor.

Figure 7: LCD Display Showing a Correct Answer.

```
What was Martin Luther King Jr famous speech commonly referred to as?
a) I have a dream
b) Freedom Fighters
c) Dead and Evil
d) Rise Up
```

Figure 8: Second Question Displayed on Serial Monitor.

Figure 9: LCD Display Showing an Incorrect Answer.

4.3 End of Game and Restart Mechanism

At the end of the 10 questions, the game provides the final score and the possibility to restart the game (by pressing SELECT).

Figure 10: LCD Display Showing an Incorrect Answer.

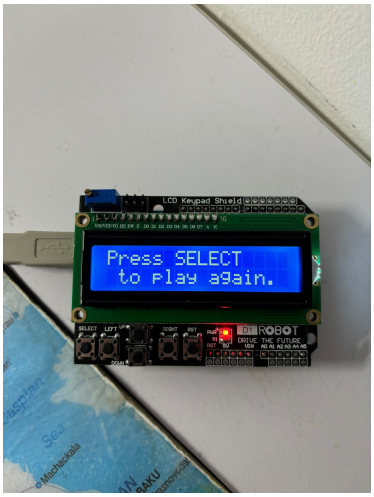

Figure 11: LCD Display Showing an Incorrect Answer.

5 Link to Source Code

The complete source code for the Trivia Game project is available in a GitHub repository.

Repository Details:

- **URL:** `https://github.com/BobosRazvan/Arduino-Platform-Trivia-Project`

6 Project Summary

The Trivia Game project was designed to be an entertaining way to answer trivia questions. Here are the key points of the project:

- **Objective:** To ask players questions on various topics.
- **Technologies Used:**
  - Arduino Uno Board and D1 Robot Shield for hardware interfacing.
  - HD44780 LCD for display.
  - Embedded C++ for software development.
- **Features:**
  - Multiple choice questions with feedback on answers.
  - Serial communication for question delivery.
  - Score tracking.